Real-time 3D-aware Portrait Editing from a Single Image

Qingyan Bai[1,2], Zifan Shi[1], Yinghao Xu[3], Hao Ouyang[1,2],
Qiuyu Wang[2], Ceyuan Yang[4], Xuan Wang[2],
Gordon Wetzstein[3], Yujun Shen[*2], and Qifeng Chen[*1]

[1] The Hong Kong University of Science and Technology
[2] Ant Group
[3] Stanford University
[4] Shanghai AI Laboratory

Abstract. This work presents **3DPE**, a practical method that can efficiently edit a face image following given prompts, like reference images or text descriptions, in a 3D-aware manner. To this end, a lightweight module is distilled from a 3D portrait generator and a text-to-image model, which provide prior knowledge of face geometry and superior editing capability, respectively. Such a design brings two compelling advantages over existing approaches. First, our method achieves *real-time* editing with a feedforward network (*i.e.*, ~0.04s per image), over 100× faster than the second competitor. Second, thanks to the powerful priors, our module could focus on the learning of editing-related variations, such that it manages to handle various types of editing simultaneously in the training phase and further supports *fast adaptation* to user-specified customized types of editing during inference (*e.g.*, with ~5min fine-tuning per style). Project page can be found here.

Keywords: 3D-aware portrait · Efficient editing

1 Introduction

Inferring the geometry and appearance from a single-view portrait image has become mature and practical [22, 37, 42, 66, 70, 74, 76], largely attributed to the utilization of priors in various 2D/3D generative models. However, only performing geometry reconstruction is insufficient. The significance of 3D portrait editing, driven by user intentions, and the need for streamlined efficiency in the editing process has been steadily increasing. This is particularly crucial in real-world applications such as AR/VR, 3D telepresence, and video conferencing, where real-time editing is often essential. Consequently, a key question arises: How can we effectively address the challenge of attaining high-fidelity portrait editing while ensuring real-time efficiency?

Traditional methods for 3D portrait editing [17] typically rely on template facial models [6, 51], which has limitations in handling the substantial geometry
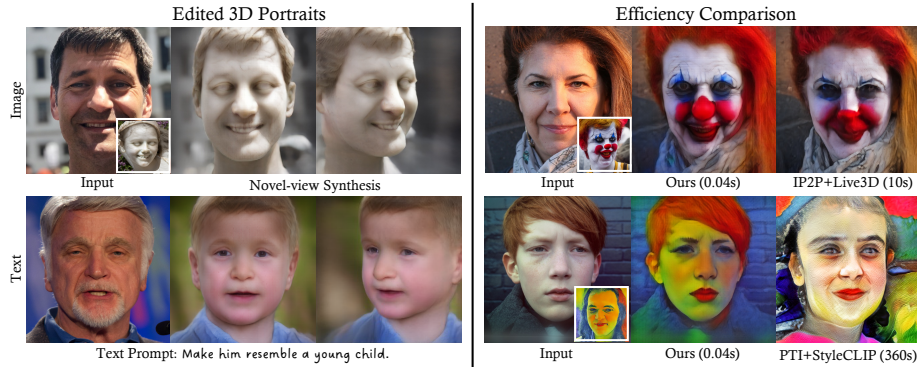
---

[*] Corresponding authors.

Fig. 1: **Photorealistic editing results** produced by our proposed **3DPE**, which allows users to perform 3D-aware portrait editing using image or text prompts. In comparison with baseline methods, such as InstructPix2Pix (IP2P) [7]+Live3D [70] and PTI [55]+StyleCLIP [50] (details are illustrated in Sec. 4), our approach accurately follows the guidance from reference prompts and maintains sufficiently better efficiency.

changes because it often overlooks precise features like hair and beard. Recent 3D GANs [13, 14, 58, 62, 75] show remarkable capabilities in generating high-fidelity 3D portraits. They can serve as powerful generative priors for 3D portrait editing when coupled with GAN inversion techniques [2, 3, 61, 86, 87]. However, these methods may encounter issues related to geometry distortion or exhibit slow speed, and the editing is constrained by the limited latent attributes. Besides, 2D diffusion models can offer a strong prior with Score Distillation Sampling (SDS) loss [52] for editing purposes [30]. Nonetheless, they often require step-by-step optimization and become a major bottleneck for real-time applications.

To this end, we present **3DPE**, a real-time **3D**-aware **P**ortrait **E**diting method driven by user-defined prompts. As shown in Fig. 1, when provided with a single-view portrait image, our approach empowers a versatile range of editing styles with one model through flexible instructions, including images and texts. We leverage the powerful 3D prior from a 3D-aware face generator and achieve a high-fidelity 3D reconstruction of the portrait image. Subsequently, we distill editing knowledge from a text-guided image editing model into a lightweight module integrated with the 3D-aware generator. This module, characterized by its minimal computational cost, allows our method to maintain real-time inference and excel in handling various types of editing, while ensuring good 3D consistency. An additional advantage is that our model supports customization through user-specified prompts with fast adaptation speed. This empowers users to build their own editing model at a minimal cost, enabling our method to cater to a broader audience.

Our method achieves real-time 3D-aware portrait editing through the utilization of a feedforward network, with a processing time of 40ms on a standard consumer GPU. Additionally, we present a comprehensive evaluation of

our method using various prompts both quantitatively and qualitatively. Our design choices are also validated through comparisons with ablated variants of our method. Compared with baseline methods, our approach demonstrates superior 3D consistency, precise texture alignment, and a substantial improvement in inference time, as reflected by the evaluation metrics. We demonstrate the versatility of our method by showcasing its capacity to perform a wide variety of edits, including text and image prompts, on portrait images.

In summary, the contributions of our work include:

- We propose a lightweight module to distill knowledge from 3D GANs and diffusion models for 3D-aware editing from a single image. Due to the minimal cost of the new module, our model maintains real-time performance.
- Our model supports fast adaptation to user-specified editing, requiring only 10 image pairs and 5 minutes for the adaptation.
- Our framework can accommodate various control signals, including text and image prompts.

2   Related Work

2.1   Generative Face Priors

Generative models aim at modeling the underlying distribution of the data, containing a wealth of prior knowledge. Recently, 3D GANs [13, 14, 25, 47, 49, 58, 59, 63–65, 69, 71, 77] are mostly adopted to learn 3D faces from single-view image dataset. The interior rich domain-specific geometry priors enable various face-related applications such as image editing [9, 66, 67, 70, 79, 85] and domain adaptation [1,34,36,46,82]. Large-scale diffusion models [56], in contrast, encode knowledge of huge datasets and thus can provide general prior information. Such priors are broadly leveraged for tasks such as image editing [7, 10, 24, 45, 83], customization [31,38,44,57], and video editing [11,12,23,43,48,53,73,80]. In our approach, we capitalize on the advantages of both the geometry prior derived from 3D GANs and the broader editing prior offered by large-scale text-to-image diffusion models, instead of relying solely on a single type of prior.

2.2   Portrait Editing from a Single Image

Although many methods work well for reconstructing [22, 26] or generating [13] faces, editing has become a necessary interface to connect these methods with real-world applications. Previously, portrait editing given a single-view image was mostly completed in 2D image space facilitated with GAN inversion [2, 3, 5, 28, 55, 60, 61, 78, 86], which enables fast editing by exploring the trajectories in the GAN's latent space. Recent diffusion-based portrait editing [7, 83] can support various editing types with texts as guidance. However, most of them are done in the 2D space, and thus there is no guarantee for the underlying 3D consistency. Therefore, 3D-aware portrait editing is crucial to achieve the goal. Some methods [15, 33, 39, 42, 66, 67, 74, 76, 81] rely on the latent space

of 3D GANs to perform editing through walking in the latent space, but are limited by the number of editable latent attributes. A few methods, such as Instruct-NeRF2NeRF [27], ClipFace [4], and LENeRF [32], attempt to leverage the large-scale models [54, 56] to edit on 3D representations with texts as guidance but require heavy iterative refinement of the edited results. Similarly, InstructPix2NeRF [40] leverages text-to-image models and text prompts to edit within the latent space of the GAN-based 3D generator. However, it is held back by the multi-step diffusion inference and the heavy structure of a GAN-based generator. Ours, in contrast, benefits from the lightweight design of the module and achieves real-time editing with ease. With such a design, our model can also adapt fast to the user-defined editing prompts. Moreover, texts cannot always illustrate the desired effects precisely, and sometimes an image prompt serves as better guidance for image editing [20, 57]. Our method can support not only text prompts but also image prompts for editing, providing a more flexible and user-friendly interface.

3   Method

Our approach takes a portrait image $\mathbf{I}$ and its camera pose $\mathbf{c}$, obtained via the face pose estimator [18]. Additionally, the referenced prompts $\mathbf{P}$ (images or texts) serve as the editing instructions. The outcome of our model is an edited version of the 3D portraits characterized by NeRF in accordance with the prompts. The edited image is denoted as $\mathbf{I}_p$. Within our framework, we achieve this objective by leveraging the knowledge of 3D GANs and a text-guided image editing model, which is distilled into a lightweight module. Following distillation, our method enables real-time 3D portrait editing and efficient adaptation to user-specified prompts. In Sec. 3.1, we offer an illustration of 3D GANs and diffusion models. The process of distilling these priors into a lightweight module is outlined in Sec. 3.2. Finally, the details for model training and inference, along with the fast adaptation for customized prompts, are presented in Sec. 3.3.

3.1   Preliminary

3D GAN Prior for Portrait Reconstruction. The 3D-aware GANs showcase the ability to synthesize photorealistic 3D images using a collection of single-view images. Notably, EG3D [13] introduces an efficient triplane 3D representation, demonstrating high-quality 3D-aware image rendering. Once trained, the generator of EG3D can be applied for single-image 3D reconstruction via GAN inversion. However, existing 3D GAN inversion methods often encounter geometry distortion or exhibit slow inference speed. To tackle these challenges, we utilize Live3D [70], a state-of-the-art single-image portrait 3D reconstruction model built upon EG3D, preserving geometry quality while ensuring real-time performance. Live3D employs a two-branch encoder, $\mathrm{E}_{high}(\cdot)$ and $\mathrm{E}_{low}(\cdot)$, to extract different resolution features from the input. It then utilizes a ViT-based
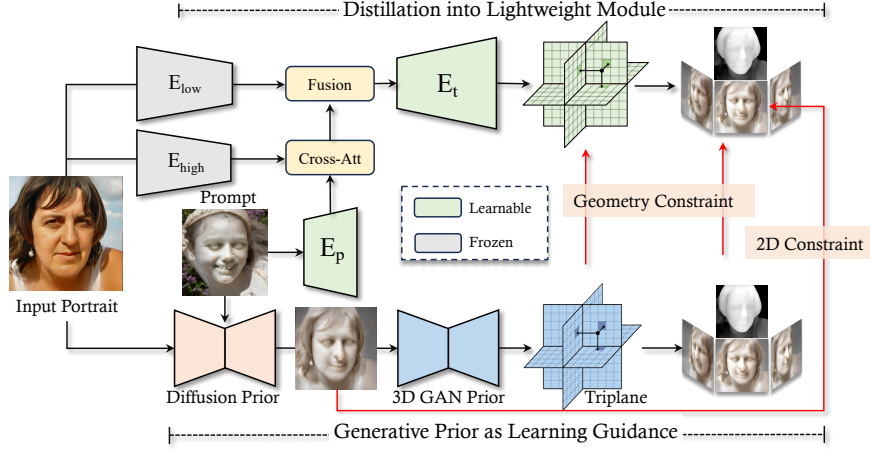
Fig. 2: Overview of our method. We distill the prior in the diffusion model and 3D GAN for real-time 3D-aware editing. Our approach is fine-tuned from Live3D [70], where we extract features from the input portrait I using $E_{high}(\cdot)$ and $E_{low}(\cdot)$. The prompt embedding is generated with $E_p(\cdot)$ and injected with the input features from $E_{high}(\cdot)$ through a cross-attention mechanism. Our model is trained to mimic the output from the diffusion prior to acquire editing knowledge and enforce geometry constraints through triplane, multi-view images, and depth supervision from the 3D prior. In this context, InstructPix2Pix [7] and Live3D serve as the diffusion and 3D prior, respectively. It's noteworthy that only $E_p(\cdot)$ and $E_t(\cdot)$ are learnable during training, while all other parameters remain frozen.

$E_t(\cdot)$ decoder [19] to transform the fused encoder features into the triplane **T**:

$$\mathbf{T} = E_t(E_{high}(\mathbf{I}), E_{low}(\mathbf{I})). \tag{1}$$

This triplane **T** is subsequently used in conjunction with the volume rendering module and upsampler of EG3D to generate photorealistic view synthesis given the camera pose c. For clarity, we use $R(\cdot)$ to denote this process.

Diffusion Prior for Portrait Editing. Only performing 3D reconstruction is insufficient as our goal is to edit portrait images. Large-scale diffusion models [56], trained on vast text-image pairs, can synthesize realistic photos with text input, offering powerful editing prior. Despite their effectiveness for face images, the editing process is often slow due to step-by-step optimization or multiple inferences in diffusion models. The scarcity of high-quality and multi-view consistent paired data significantly hinders the practical application of 3D-aware editing from a single image, which is hard to acquire with diffusion models. Therefore, our goal is to distill editing knowledge from the diffusion model, integrate it with the 3D prior in Live3D into a lightweight module, and employ it for real-time portrait editing.
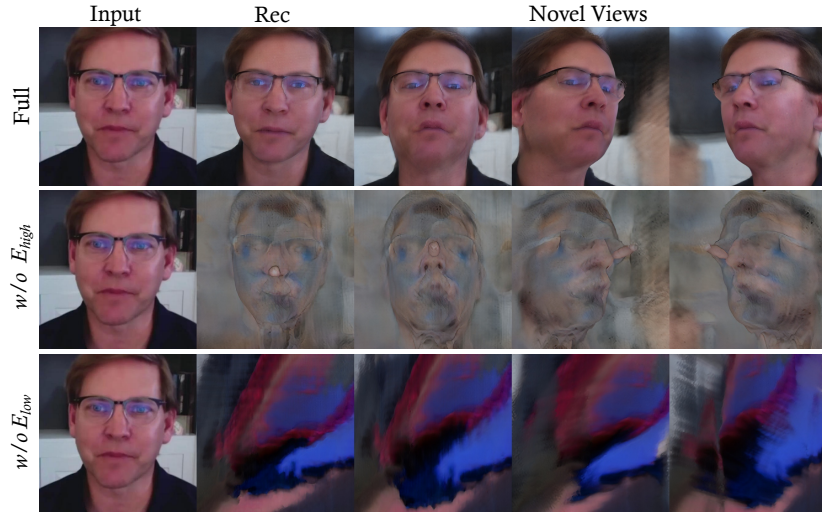
Fig. 3: Disentanglement in Live3D features. We separately disable the features from $E_{high}(\cdot)$ and $E_{low}(\cdot)$ to infer the reconstructed image. Without $E_{high}(\cdot)$, the output retains the coarse structure but loses its appearance. Conversely, when $E_{low}(\cdot)$ is deactivated, the reconstructed portraits preserve the texture (such as the blue and purple reflection on the glasses) but fail to capture the geometry.

3.2   Distilling Priors into a Lightweight Module

Our method aims to perform real-time 3D-aware editing for various prompts for the single-view portrait. Thus, it needs powerful 3D knowledge for geometry reconstruction and editing prior to handling various control signals. We leverage the strengths of both 3D GANs and diffusion models. In the following, we provide detailed presentations on how the knowledge of these two types of models is distilled into a lightweight module.

Feature Representation in Live3D. We carefully study the Live3D model and discover that, as a two-branch triplane-based 3D reconstruction model, the low-resolution and high-resolution features from the Live3D encoder $E_{low}(\cdot)$ and $E_{high}(\cdot)$ tend to learn various levels of information without explicit guidance. We conduct a study by disabling one of the branches and inferring the reconstructed images. As illustrated in Fig. 3, when the high-resolution encoder $E_{high}(\cdot)$ is disabled with random weights, the inference image retains a similar structure but loses its detailed appearance. Conversely, when the low-resolution encoder $E_{low}(\cdot)$ is disabled, the reconstructed portraits preserve some of the texture from the input but struggle to capture the geometry. Based on this analysis, the features of the two branches separately model low-frequency and high-frequency information. This insight motivates our model design in terms of how to distill knowledge from the diffusion model and 3D GANs.

Geometry Prediction for Inputs I. In our framework, we use the prompts **P** to refine the input portrait **I** into the edited image $\mathbf{I}_p$. The $\mathbf{I}_p$ inherits a similar structure to the input **I**, providing coarse geometry or structural guidance in the editing process. As discussed in the above section, the encoder $\mathrm{E}_{low}(\cdot)$ consistently generates low-frequency features to represent geometry cues, making it well-suited for preserving structured information in the input portrait. As a result, we freeze $\mathrm{E}_{low}(\cdot)$ and leverage it to produce structure features $\mathbf{F}_g = \mathrm{E}_{low}(\mathbf{I})$ for coarse geometry prediction.

Injecting Prompts P as Condition. In contrast to input images, prompts typically offer more high-frequency and texture information to guide and control the editing process. Accordingly, it is appropriate to incorporate the prompts into the high-level branch $\mathrm{E}_{high}(\cdot)$ in Live3D to extract the high-frequency features. To incorporate prompts, we utilize the prompt encoder $\mathrm{E}_p(\cdot)$ to create prompt embeddings that are fused into our model through cross attention, which is similar to the strategy in Stable Diffusion [56]. Specifically, we add a cross-attention layer after $\mathrm{E}_{high}(\cdot)$ to obtain the feature updated with the prompt embeddings:

$$\mathbf{F}_a = \mathrm{CrossAttention}(\mathrm{E}_{high}(\mathbf{I}), \mathrm{E}_p(\mathbf{P})), \tag{2}$$

where the encoder $\mathbf{E}_p(\cdot)$ is a transformer (MAE [29] for image and CLIP [54] for text). The $\mathrm{E}_{high}(\mathbf{I})$ serves as the query and $\mathrm{E}_p(\mathbf{P})$ is employed as key and value. The updated feature $\mathbf{F}_a$ and the geometry feature $\mathbf{F}_g$ are then feed into the decoder $\mathrm{E}_t(\cdot)$ to infer the triplane $\mathbf{T_p}$:

$$\mathbf{T_p} = \mathrm{E}_t(\mathbf{F}_a, \mathbf{F}_g). \tag{3}$$

Distilling Diffusion and 3D GAN Prior. Unlike previous methods that require large amounts of 3D data, our model operates with only 2D paired data generated by the 2D editing model [7]. For both visual and textual prompts, we generate edited images $\mathbf{I}_{gt}$ with the diffusion-based image editing model [7] as the pseudo labels. The details of data generation can be found in Sec. 4.1.

With the predicted triplane $\mathbf{T_p}$, we can render the image using the pretrained EG3D model and then calculate the reconstruction loss as follows:

$$\mathcal{L}_{2d} = \ell_{\mathbf{I}}(\mathbf{I}_{gt}, \mathrm{R}(\mathbf{T_p}, \mathbf{c})), \tag{4}$$

where $\mathrm{R}(\cdot)$ is the rendering module of EG3D, $\mathbf{c}$ is the camera pose of input portrait, and $\ell_1(\cdot)$ is an image reconstruction loss penalizing the difference between the ground truth $\mathbf{I}_{gt}$ and the rendering $\mathbf{I}_p = \mathrm{R}(\mathbf{T_p}, \mathbf{c})$). It's important to note that $\mathcal{L}_{2d}$, used to reconstruct the $\mathbf{I}_{gt}$, essentially distills the knowledge of editing from the diffusion model.

We observe that $\mathcal{L}_{2d}$ helps the model to reconstruct well on the input camera view but suffers from geometry distortion during novel-view synthesis. To fully exploit the 3D properties of Live3D, we propose to distill the 3D knowledge of Live3D. Specifically, we leverage the pretrained Live3D to infer the triplane $\mathbf{T}_{gt}$, and multi-view depths $\mathcal{D}_{gt}$ and images $\mathcal{I}_{gt}$ of the pseudo-label image $\mathbf{I}_{gt}$:

$$\mathbf{T}_{gt}, \mathcal{D}_{gt} = \mathrm{G}(\mathbf{I}_{gt}, \mathcal{C}), \tag{5}$$

where $G(\cdot)$ represents the inference process of Live3D, where $\mathcal{C} = \{\mathbf{c}_1, .., \mathbf{c}_n\}$ is the camera set, and n denotes the number of cameras. We also render the multi-view images $\mathcal{I}_p$ and depths $\mathcal{D}_p$ of the edited image I_p from the triplane $\mathbf{T}_p$ and then define the objective:

$$\mathcal{L}_{3d} = \ell_{\mathcal{I}}(\mathcal{I}_p, \mathcal{I}_{gt}) + \ell_{\mathbf{T}}(\mathbf{T}_p, \mathbf{T}_{gt}) + \ell_{\mathcal{D}}(\mathcal{D}_p, \mathcal{D}_{gt}), \tag{6}$$

where $\ell_{\mathcal{I}}(\cdot)$, $\ell_{\mathbf{T}}(\cdot)$ and $\ell_{\mathcal{D}}(\cdot)$ is the reconstruction loss penalizing the difference in image, triplane, and depth between I_p and I_{gt}.

3.3   Training and Inference

Training. During the training phase, we sample a triplet consisting of the input portrait $\mathbf{I}$ along with its camera pose $\mathbf{c}$, prompts $\mathbf{P}$, and the pseudo-label image $\mathbf{I}_{gt}$ generated by the 2D diffusion model. We leverage the Live3D model as a pretrained model and add an additional prompts encoder E_p to extract prompt embeddings. The overall learning objective can be described as follows:

$$L = \lambda_1 L_{2D} + \lambda_2 L_{3D}, \tag{7}$$

where λ_1 and λ_2 are loss weights. In our setting, λ_1 and λ_2 are both set to 1.0. For the reconstruction loss, $\ell_{\mathcal{I}}(\cdot)$, $\ell_{I}(\cdot)$ are combinations of L_2 loss and LPIPS loss [84], with loss weights being 1 and 2, respectively. $\ell_{\mathcal{D}}(\cdot)$ and $\ell_{T}(\cdot)$ are L_1 loss. Notably, based on the study on Live3D feature representation, during training only $E_p(\cdot)$, $E_t(\cdot)$ are set to be learnable, while other modules are frozen. It allows our model to leverage Live3D knowledge as much as possible and converge at a very fast speed.

Inference. For inference, users can provide a single 2D portrait image and choose an image prompt or text prompt. Our model is able to generate the edited 3D NeRF along with photorealistic view synthesis.

Customized Prompts Adaptation. To accommodate customized styles provided by users, we propose a method to adapt our pretrained encoder to user-defined styles. We increase the tuning efficiency by optimizing only $E_p(\cdot)$ and the normalization layers in $E_t(\cdot)$ with the same learning objective Eq. 7. This method allows us to limit the training data to only 10 image pairs and the learning time to 5 minutes on a single GPU.

4   Experiments

4.1   Experimental Setup

Training Settings. For real faces, we adopt the FFHQ dataset [35] at 512×512 resolution with camera parameters aligned by EG3D [13]. In order to obtain the stylized images as pseudo labels, and the corresponding textual and visual prompts, we leverage the text-guided editing model InstructPix2Pix (IP2P) [7] to

Fig. 4: Qualitative comparisons. We compare the results of several baselines with image prompts and text prompts. In each case, we include the edited portraits as well as their novel view renderings. Our method generates high-quality edited portraits with better 3D quality and alignment with the referenced prompts.

edit the real faces. Specifically, we manually generate and select textual prompts, and send them to IP2P together with the real images to obtain the edited 2D images. The text prompts of our training and testing set are obtained by augmenting the aforementioned textual prompts with the large language model GPT [8] following IP2P [7]. While the image prompts of our training and testing set are picked from the edited 2D images. We conduct experiments on 20 styles, and for each style, we synthesize 1000 images, resulting in 20,000 image pairs with text prompts for model training. For each style, we use 8 textual prompts for training and 5 textual prompts for testing. We adopt a learning rate of 5e-5 and optimize the model for 60k iterations with a batch size of 32. The entire training procedure is completed over a period of 40 hours using 8 NVIDIA A100 GPUs.

Baselines. We conduct comparisons against two categories of methods that achieve analogous outcomes: 1) 3D reconstruction coupled with 3D editing; 2) 2D editing followed by 3D reconstruction. For the first category, we implement three baseline approaches, wherein we firstly leverage PTI [55] or Live3D [70] for reconstruction and edit both of them following StyleCLIP [50]. Specifically, for experiments in PTI + StyleCLIP, we first conduct pivotal tuning [55] based on the predicted latent codes predicted by e4e [68] to reconstruct the given image in

Table 1: Quantitative comparisons. We compare several baselines on the 100 images of FFHQ dataset. It's important to note that we exclude $CLIP_r$ for PTI+StyleCLIP and Live3D+StyleCLIP since these models utilize CLIP for optimization. Our model excels in 3D quality and achieves a remarkable 250x improvement (compared with IP2P+Live3D) in inference speed, achieving real-time performance.

Method	$ID_t \uparrow$	$CLIP_r \uparrow$	3D$\uparrow$	Time$\downarrow$
PTI+StyleCLIP	0.11	-	0.73	360s
Live3D+StyleCLIP	**0.63**	-	0.72	30s
IP2P+Live3D	0.44	0.62	0.75	10s
Ours	0.47	**0.73**	**0.76**	**0.04s**

a 3D-aware manner. Then we perform editing on the latent codes following [50], where CLIP loss [54] is utilized between the model output and the prompt (no matter whether it is an image or text). The setting of Live3D + StyleCLIP is similar to PTI + StyleCLIP except that we obtain the triplane by Live3D and perform editing on this 3D representation. SPI [81] + StyleGAN-NADA [21] (StyleNADA for the abbreviation in Fig. 4) serves as the third baseline in the first category, where SPI is used for 3D reconstruction and StyleGAN-NADA is utilized to optimize the 3D generator for editing. In the second category, we employ a diffusion-based text-guided 2D image editing method IP2P [7] prior to executing 3D reconstruction via Live3D. IP2P being a text-instructed editing model, we need to first extract text prompts from the reference image with the image caption algorithm BLIP [41] when image prompts are input. The 2D edited results could then be obtained by sending the input image and the text prompt to IP2P. The final 3D results are acquired by inverting the 2D editing results with Live3D.

Evaluation Criteria. We evaluate all metrics on 100 pairs of images processed from FFHQ. We conduct a comprehensive evaluation of the editing performance in the following four aspects: identity preservation, reference alignment, 3D consistency, and inference speed. *1) Identity preservation (ID_t)* aims to measure the preservation of the original identity by calculating the cosine similarity between the identity feature of the input image **I** and that of the edited image $\mathbf{I}_p$. We use the ArcFace model [16] to extract identity features. *2) Reference alignment ($CLIP_r$)* targets at assessing the alignment of the output editing styles to the desired input prompt by computing the cosine similarity in the CLIP [50] feature space. *3) 3D consistency (3D)* on the edited outputs is measured following the evaluation protocols established by EG3D [13]. This involves calculating the identity similarity across multiple views. *4) Inference speed (Time)* is measured on a single NVIDIA A6000 GPU with an average of 100 samples.
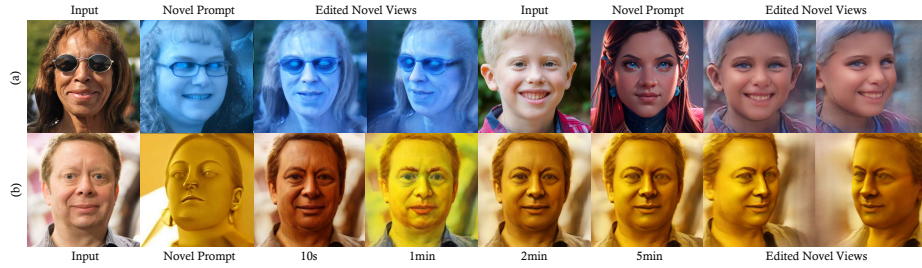
Fig. 5: (a) Generated results of customized prompt adaptation and (b) its learning process. We show the intermediate testing results at 10s, 1min, 2min and 5min during adaptation for the novel style golden statue.
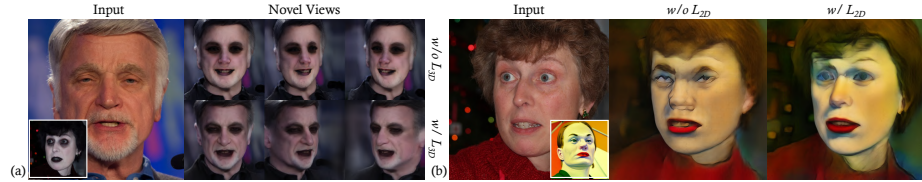

Fig. 6: Qualitative comparison for ablations on (a) the distillation loss of 3D GAN ($\mathcal{L}_{3D}$) and (b) diffusion models ($\mathcal{L}_{2D}$).

4.2 Efficient 3D-aware Portrait Editing

We make an in-depth analysis of our efficient portrait editing method both quantitatively and qualitatively. The results are included in Tab. 1 and Fig. 4. For reference alignment, we do not report $CLIP_r$ for methods that leverage CLIP for optimization since it is evaluated with CLIP as well. The standout feature of our method is its efficiency. Our method achieves an inference speed of merely 40ms, which improves over 100 times compared to the fastest existing baselines, which require around 10 seconds. Because of the efficient knowledge distillation, our approach is also good in preserving the identity, adhering more closely to the reference prompts and achieving the best 3D consistency.

As in Fig. 4, compared with PTI + CLIP methods, our model enables precise alignment with inputs and prompts, while the baseline always generates low-fidelity textures, and the resulting geometry has many artifacts. Although Live3D + CLIP achieves the best ID_t score, the edited results are basically unchanged compared to the input. We suspect that Live3D does not possess a latent space and loses the editing priors, making it challenging for the results of CLIP optimization to align well with prompts. SPI + StyleGAN-NADA performs relatively slowly (18 minutes per instance) because both the 3D reconstruction and the editing are achieved by optimizing the generator. Although it can produce realistic textures, it fails to reconstruct fine details and to edit following the prompts. In contrast to the 2D editing and subsequent 3D reconstruction

Table 2: Ablation study on the number of styles used for pretraining. We report LPIPS score after 0.1min, 1min, 2min, and 5min fine-tuning for evaluation.

#Style/Time	0.1min	1min	2min	5min
3	0.7758	0.6370	0.5312	0.5220
10	0.5932	0.5734	0.5106	0.5012
20	0.5703	0.5290	0.5022	0.4849

Table 3: Ablation study on the number of data pairs used for novel prompt adaptation. We report LPIPS score after 0.1min, 1min, 2min, and 5min fine-tuning for evaluation.

#Pair/Time	0.1min	1min	2min	5min
2	0.6191	0.5431	0.5213	0.5212
5	0.6148	0.5324	0.5064	0.4971
10	0.6076	0.5242	0.4983	0.4935
20	0.5981	0.5525	0.4954	0.4895
50	0.5891	0.5718	0.4922	0.4797

pipeline (IP2P + Live3D), our method produces textures more consistent with the prompt and aligns better with the input structure.

4.3   Adaptation for Customized Editing

With the trained editing network, our method already supports a variety of 3D-aware editing styles. Following DreamBooth [57] and to expand the selection and better conform to users' preferences, we offer an efficient method for fast adaptation to the user-defined customized prompts. Users can personalize the editing network by providing a modest set of 10 reference editing pairs. These pairs can be either handpicked from artist-created examples or produced using text-guided image editing models. As described in Sec. 3.3, only the partial of the encoder would be optimized. The adaptation process itself is remarkably fast, requiring only about 5 minutes to accomplish the learning of the customized prompts. Please also note that, once the user-defined style has been trained, all the incoming testing results can be obtained in a real-time manner. In Fig. 5, we present the testing results and learning process of the adaptation. Our model can quickly master the customized knowledge in about 2 minutes. With further training (*e.g.*, 5 minutes), the edited results become more stylized, demonstrating a trade-off between authenticity and stylization. Upon completion of the adaptation, the method allows users to edit testing inputs in these newly learned styles with a minimal inference time of 0.04s. This rapid performance indicates that the method is well-optimized for real-time applications, providing a seamless and efficient user experience.

4.4   Ablation Studies

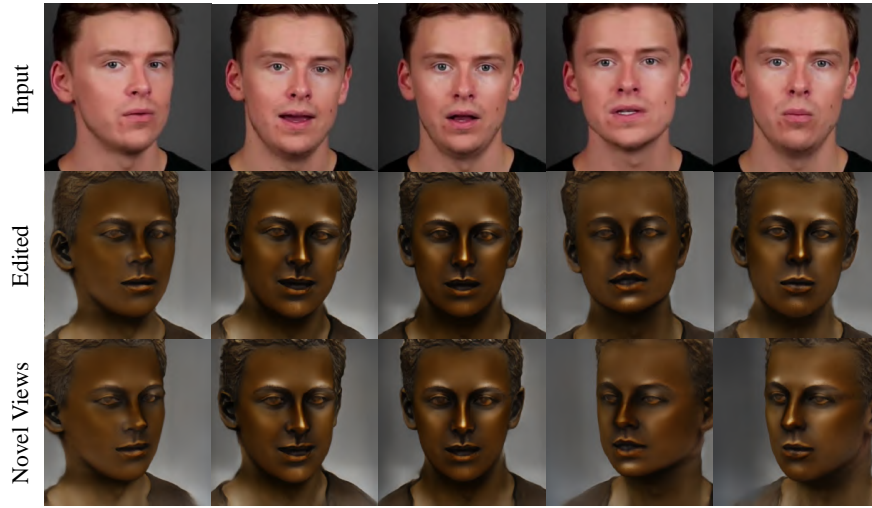We analyze our model and validate our design choice by ablating the components.

Fig. 7: Video editing results with 3DPE. We use text prompts "Make him a bronze statue" to edit the input video. Our method can accurately reconstruct and preserve challenging facial expressions and achieve high-quality novel view renderings.

$\mathcal{L}_{3D}$ **for Distillation of Live3D**. We compare our model trained with and without $\mathcal{L}_{3D}$. As shown in Fig. 6, we observe that the model without $\mathcal{L}_{3D}$ results in very flat geometry, and the synthesis of the side view loses the normal structure with many artifacts. This highlights the importance of the 3D knowledge from Live3D in training our model with 2D paired data, enabling the inference of reasonable face geometry without relying on any 3D data sources.

$\mathcal{L}_{2D}$ **for Distillation of Diffusion Models**. We also study the effect of $\mathcal{L}_{2D}$, which represents using 2D editing priors from the diffusion model [7] to optimize the encoder. From Fig. 6, we observe that $\mathcal{L}_{2D}$ is critical for our model, especially in preserving detailed texture. Without it, the edited image tends to lose the appearance information from prompts and exhibits structural artifacts around the nose.

Ablation on Pretraining Data Scale for Customized Prompt Adaptations. In this part, we conduct ablation studies on the pretraining scale and investigate its influence on customized propmts. Specifically, we choose various 3DPE models pretrained with different style amounts (3, 10, and 20 styles respectively) and perform novel prompt adaptation on a single A6000 GPU. We choose the golden statue as the customized style and report LPIPS score between the predicted stylized images and the pseudo stylized labels produced by IP2P [7]. Experimental results in Tab. 2 reveal that the proposed method could benefit from the improving pretraining data scale and adapt faster and eventually perform better with more data, which suggests the potential of the model for future scaling up.

Ablation on Tuning Data Scale for Customized Prompt Adaptations.
To investigate the ability of our method to adapt to user-defined novel prompts,
we conduct an ablation study on the number of novel-prompt data pairs required
for training. Similar to the prior experiments on the pretraining scale, we perform
adaptation with 2, 5, 10, 20, or 50 pairs of novel-prompt data on a single A6000
GPU. We experimentally find that our model can converge in about 5 minutes,
and therefore, we report the perceptual distance between the model output and
the provided ground truth at 0.1min, 1min, 2min, and 5min for comparison. As
shown in Tab. 3, our model can adapt to the new style in around 2 minutes. As
the number of provided data pairs increases, the model can adapt faster at the
beginning.

4.5   Applications

One application of our pipeline is to edit videos, requiring accurate reconstruc-
tion and efficient editing simultaneously. We demonstrate the effectiveness of
our pipeline on talking face videos, as illustrated in Fig. 7. In this experiment,
we perform sequence editing in a frame-by-frame manner by sending an image
sequence from TalkingHead-1KH [72] dataset together with the text prompt
"Make him a bronze statue" to 3DPE. The three rows in Fig. 7 respectively
illustrate the input frames, the edited frames, and the novel views. Our method
accurately reconstructs facial expressions, while achieving high-quality editing
results. Since our method is 3D-aware, we can naturally view videos from novel
perspectives. To better experience our method, we design an interactive system
that allows users for interactive editing. Users are required to provide the image
or text prompt to indicate the desired style and input an input image that is to
be edited, and the system will apply our method on the given image and prompt
to output the edited image as well as novel-view edited results.

5   Discussion

Limitations. Despite achieving state-of-the-art performance in quality and ef-
ficiency, our method exhibits inconsistencies in details for novel-view rendering
because the EG3D framework relies on a super-resolution module. Additionally,
when our method is applied to video editing, it presents flickering artifacts since
our model is originally designed for image-based editing.
Conclusion. In this work, we introduce 3DPE for real-time 3D-aware portrait
editing. By distilling the powerful knowledge of diffusion models and 3D GANs
into a lightweight module, our model significantly reduces editing time while
ensuring quality. Benefiting from the distillation process, 3DPE could handle
multiple styles in a single model and perform both image-guided and text-guided
editing. The advantages of our method empower us to generate photorealistic 3D
portraits, a capability crucial for the visual effects industry, AR/VR systems, and
teleconferencing, among other applications, while we firmly oppose any misuse
that could compromise privacy and security.

Acknowledgements

References

1. Abdal, R., Lee, H.Y., Zhu, P., Chai, M., Siarohin, A., Wonka, P., Tulyakov, S.: 3DAvatarGAN: Bridging domains for personalized editable avatars. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
2. Abdal, R., Qin, Y., Wonka, P.: Image2StyleGAN: How to embed images into the StyleGAN latent space? In: Int. Conf. Comput. Vis. (2019)
3. Abdal, R., Qin, Y., Wonka, P.: Image2StyleGAN++: How to edit the embedded images? In: IEEE Conf. Comput. Vis. Pattern Recog. (2020)
4. Aneja, S., Thies, J., Dai, A., Nießner, M.: ClipFace: Text-guided editing of textured 3d morphable models. In: ACM SIGGRAPH Conf. Proc. (2023)
5. Bai, Q., Xu, Y., Zhu, J., Xia, W., Yang, Y., Shen, Y.: High-fidelity GAN inversion with padding space. In: Eur. Conf. Comput. Vis. (2022)
6. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Seminal Graphics Papers: Pushing the Boundaries (2023)
7. Brooks, T., Holynski, A., Efros, A.A.: InstructPix2Pix: Learning to follow image editing instructions. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
8. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: Adv. Neural Inform. Process. Syst. (2020)
9. Cai, S., Obukhov, A., Dai, D., Van Gool, L.: Pix2NeRF: Unsupervised conditional p-gan for single image to neural radiance fields translation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
10. Cao, M., Wang, X., Qi, Z., Shan, Y., Qie, X., Zheng, Y.: MasaCtrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In: Int. Conf. Comput. Vis. (2023)
11. Ceylan, D., Huang, C.H.P., Mitra, N.J.: Pix2Video: Video editing using image diffusion. In: Int. Conf. Comput. Vis. (2023)
12. Chai, W., Guo, X., Wang, G., Lu, Y.: StableVideo: Text-driven consistency-aware diffusion video editing. In: Int. Conf. Comput. Vis. (2023)
13. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L.J., Tremblay, J., Khamis, S., et al.: Efficient geometry-aware 3d generative adversarial networks. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
14. Chan, E.R., Monteiro, M., Kellnhofer, P., Wu, J., Wetzstein, G.: pi-GAN: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In: IEEE Conf. Comput. Vis. Pattern Recog. (2021)
15. Chang, S., Kim, G., Kim, H.: HairNeRF: Geometry-aware image synthesis for hairstyle transfer. In: Int. Conf. Comput. Vis. (2023)
16. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: ArcFace: Additive angular margin loss for deep face recognition. In: IEEE Conf. Comput. Vis. Pattern Recog. (2019)
17. Deng, Y., Yang, J., Chen, D., Wen, F., Tong, X.: Disentangled and controllable face image generation via 3d imitative-contrastive learning. In: IEEE Conf. Comput. Vis. Pattern Recog. (2020)

18. Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., Tong, X.: Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In: IEEE Conf. Comput. Vis. Pattern Recog. (2019)
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: Int. Conf. Learn. Represent. (2021)
20. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-Or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. In: Int. Conf. Learn. Represent. (2023)
21. Gal, R., Patashnik, O., Maron, H., Bermano, A.H., Chechik, G., Cohen-Or, D.: StyleGAN-NADA: Clip-guided domain adaptation of image generators. ACM Trans. Graph. (2022)
22. Gao, C., Shih, Y., Lai, W.S., Liang, C.K., Huang, J.B.: Portrait neural radiance fields from a single image. arXiv preprint arXiv:2012.05903 (2020)
23. Geyer, M., Bar-Tal, O., Bagon, S., Dekel, T.: TokenFlow: Consistent diffusion features for consistent video editing. In: Int. Conf. Learn. Represent. (2024)
24. Graikos, A., Malkin, N., Jojic, N., Samaras, D.: Diffusion models as plug-and-play priors. In: Adv. Neural Inform. Process. Syst. (2022)
25. Gu, J., Liu, L., Wang, P., Theobalt, C.: StyleNeRF: A style-based 3d-aware generator for high-resolution image synthesis. In: Int. Conf. Learn. Represent. (2022)
26. Guo, J., Yu, J., Lattas, A., Deng, J.: Perspective reconstruction of human faces by joint mesh and landmark regression. In: Eur. Conf. Comput. Vis. (2022)
27. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-NeRF2NeRF: Editing 3d scenes with instructions. In: Int. Conf. Comput. Vis. (2023)
28. Härkönen, E., Hertzmann, A., Lehtinen, J., Paris, S.: GANSpace: Discovering interpretable GAN controls. Adv. Neural Inform. Process. Syst. (2020)
29. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
30. Hertz, A., Aberman, K., Cohen-Or, D.: Delta denoising score. In: Int. Conf. Comput. Vis. (2023)
31. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: Int. Conf. Learn. Represent. (2022)
32. Hyung, J., Hwang, S., Kim, D., Lee, H., Choo, J.: Local 3d editing via 3d distillation of clip knowledge. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
33. Jiang, K., Chen, S.Y., Fu, H., Gao, L.: NeRFFaceLighting: Implicit and disentangled face lighting representation leveraging generative prior in neural radiance fields. ACM Trans. Graph. (2023)
34. Jin, W., Ryu, N., Kim, G., Baek, S.H., Cho, S.: Dr. 3D: Adapting 3d GANs to artistic drawings. In: ACM SIGGRAPH Asia Conf. Proc. (2022)
35. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: IEEE Conf. Comput. Vis. Pattern Recog. (2019)
36. Kim, G., Chun, S.Y.: DATID-3D: Diversity-preserved domain adaptation using text-to-image diffusion for 3d generative model. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
37. Ko, J., Cho, K., Choi, D., Ryoo, K., Kim, S.: 3D GAN inversion with pose optimization. In: IEEE Winter Conf. Appl. Comput. Vis. (2023)

38. Kumari, N., Zhang, B., Zhang, R., Shechtman, E., Zhu, J.Y.: Multi-concept customization of text-to-image diffusion. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
39. Li, J., Li, J., Zhang, H., Liu, S., Wang, Z., Xiao, Z., Zheng, K., Zhu, J.: PREIM3D: 3d consistent precise image attribute editing from a single image. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
40. Li, J., Liu, S., Liu, Z., Wang, Y., Zheng, K., Xu, J., Li, J., Zhu, J.: Instruct-Pix2NeRF: Instructed 3d portrait editing from a single image. arXiv preprint arXiv:2311.02826 (2023)
41. Li, J., Li, D., Xiong, C., Hoi, S.: BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: Int. Conf. Mach. Learn. (2022)
42. Lin, C., Lindell, D., Chan, E., Wetzstein, G.: 3D GAN inversion for controllable portrait image animation. In: Eur. Conf. Comput. Vis. Workshop (2022)
43. Liu, S., Zhang, Y., Li, W., Lin, Z., Jia, J.: Video-P2P: Video editing with cross-attention control. arXiv preprint arXiv:2303.04761 (2023)
44. Liu, Z., Feng, R., Zhu, K., Zhang, Y., Zheng, K., Liu, Y., Zhao, D., Zhou, J., Cao, Y.: Cones: Concept neurons in diffusion models for customized generation. In: Int. Conf. Mach. Learn. (2023)
45. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: SDEdit: Guided image synthesis and editing with stochastic differential equations. In: Int. Conf. Learn. Represent. (2022)
46. Nitzan, Y., Gharbi, M., Zhang, R., Park, T., Zhu, J.Y., Cohen-Or, D., Shechtman, E.: Domain expansion of image generators. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
47. Or-El, R., Luo, X., Shan, M., Shechtman, E., Park, J.J., Kemelmacher-Shlizerman, I.: StyleSDF: High-resolution 3d-consistent image and geometry generation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
48. Ouyang, H., Wang, Q., Xiao, Y., Bai, Q., Zhang, J., Zheng, K., Zhou, X., Chen, Q., Shen, Y.: CoDeF: Content deformation fields for temporally consistent video processing. arXiv preprint arXiv:2308.07926 (2023)
49. Pan, X., Xu, X., Loy, C.C., Theobalt, C., Dai, B.: A shading-guided generative implicit model for shape-accurate 3d-aware image synthesis. Adv. Neural Inform. Process. Syst. (2021)
50. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D.: StyleCLIP: Text-driven manipulation of StyleGAN imagery. In: Int. Conf. Comput. Vis. (2021)
51. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3d face model for pose and illumination invariant face recognition. In: IEEE international conference on advanced video and signal based surveillance (2009)
52. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: DreamFusion: Text-to-3d using 2d diffusion. In: Int. Conf. Learn. Represent. (2023)
53. Qi, C., Cun, X., Zhang, Y., Lei, C., Wang, X., Shan, Y., Chen, Q.: FateZero: Fusing attentions for zero-shot text-based video editing. In: Int. Conf. Comput. Vis. (2023)
54. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Int. Conf. Mach. Learn. (2021)
55. Roich, D., Mokady, R., Bermano, A.H., Cohen-Or, D.: Pivotal tuning for latent-based editing of real images. ACM Trans. Graph. (2022)

56. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
57. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dream-Booth: Fine tuning text-to-image diffusion models for subject-driven generation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
58. Schwarz, K., Liao, Y., Niemeyer, M., Geiger, A.: GRAF: Generative radiance fields for 3d-aware image synthesis. In: Adv. Neural Inform. Process. Syst. (2020)
59. Schwarz, K., Sauer, A., Niemeyer, M., Liao, Y., Geiger, A.: VoxGRAF: Fast 3d-aware image synthesis with sparse voxel grids. In: Adv. Neural Inform. Process. Syst. (2022)
60. Shen, Y., Gu, J., Tang, X., Zhou, B.: Interpreting the latent space of GANs for semantic face editing. In: IEEE Conf. Comput. Vis. Pattern Recog. (2020)
61. Shen, Y., Yang, C., Tang, X., Zhou, B.: InterFaceGAN: Interpreting the disentangled face representation learned by GANs. IEEE Trans. Pattern Anal. Mach. Intell. (2020)
62. Shi, Z., Peng, S., Xu, Y., Geiger, A., Liao, Y., Shen, Y.: Deep generative models on 3d representations: A survey. arXiv preprint arXiv:2210.15663 (2022)
63. Shi, Z., Shen, Y., Xu, Y., Peng, S., Liao, Y., Guo, S., Chen, Q., Yeung, D.Y.: Learning 3d-aware image synthesis with unknown pose distribution. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
64. Shi, Z., Xu, Y., Shen, Y., Zhao, D., Chen, Q., Yeung, D.Y.: Improving 3d-aware image synthesis with a geometry-aware discriminator. In: Adv. Neural Inform. Process. Syst. (2022)
65. Skorokhodov, I., Tulyakov, S., Wang, Y., Wonka, P.: EpiGRAF: Rethinking training of 3d gans. In: Adv. Neural Inform. Process. Syst. (2022)
66. Sun, J., Wang, X., Shi, Y., Wang, L., Wang, J., Liu, Y.: IDE-3D: Interactive disentangled editing for high-resolution 3d-aware portrait synthesis. ACM Trans. Graph. (2022)
67. Sun, J., Wang, X., Zhang, Y., Li, X., Zhang, Q., Liu, Y., Wang, J.: FENeRF: Face editing in neural radiance fields. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
68. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for StyleGAN image manipulation. ACM Trans. Graph. (2021)
69. Tran, P., Zakharov, E., Ho, L.N., Tran, A.T., Hu, L., Li, H.: VOODOO 3D: Volumetric portrait disentanglement for one-shot 3d head reenactment. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)
70. Trevithick, A., Chan, M., Stengel, M., Chan, E., Liu, C., Yu, Z., Khamis, S., Chandraker, M., Ramamoorthi, R., Nagano, K.: Real-time radiance fields for single-image portrait view synthesis. ACM Trans. Graph. (2023)
71. Wang, Q., Shi, Z., Zheng, K., Xu, Y., Peng, S., Shen, Y.: Benchmarking and analyzing 3d-aware image synthesis with a modularized codebase. In: Adv. Neural Inform. Process. Syst. (2023)
72. Wang, T.C., Mallya, A., Liu, M.Y.: One-shot free-view neural talking-head synthesis for video conferencing. In: IEEE Conf. Comput. Vis. Pattern Recog. (2021)
73. Wang, W., Jiang, Y., Xie, K., Liu, Z., Chen, H., Cao, Y., Wang, X., Shen, C.: Zero-shot video editing using off-the-shelf image diffusion models. arXiv preprint arXiv:2303.17599 (2023)
74. Wang, Y., Xu, T., Wu, Y., Li, M., Chen, W., Xu, L., Yu, J.: NARRATE: A normal assisted free-view portrait stylizer. arXiv preprint arXiv:2207.00974 (2022)

75. Xia, W., Xue, J.H.: A survey on deep generative 3d-aware image synthesis. ACM Computing Surveys (2023)
76. Xie, J., Ouyang, H., Piao, J., Lei, C., Chen, Q.: High-fidelity 3d GAN inversion by pseudo-multi-view optimization. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
77. Xu, Y., Peng, S., Yang, C., Shen, Y., Zhou, B.: 3d-aware image synthesis via learning structural and textural representations. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
78. Xu, Y., Shen, Y., Zhu, J., Yang, C., Zhou, B.: Generative hierarchical features from synthesizing images. In: IEEE Conf. Comput. Vis. Pattern Recog. (2021)
79. Yang, S., Jiang, L., Liu, Z., Loy, C.C.: Pastiche master: Exemplar-based high-resolution portrait style transfer. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
80. Yang, S., Zhou, Y., Liu, Z., , Loy, C.C.: Rerender a video: Zero-shot text-guided video-to-video translation. In: ACM SIGGRAPH Asia Conf. Proc. (2023)
81. Yin, F., Zhang, Y., Wang, X., Wang, T., Li, X., Gong, Y., Fan, Y., Cun, X., Shan, Y., Oztireli, C., et al.: 3D GAN inversion with facial symmetry prior. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
82. Zhang, J., Lan, Y., Yang, S., Hong, F., Wang, Q., Yeo, C.K., Liu, Z., Loy, C.C.: DeformToon3D: Deformable neural radiance fields for 3d toonification. In: Int. Conf. Comput. Vis. (2023)
83. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Int. Conf. Comput. Vis. (2023)
84. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: IEEE Conf. Comput. Vis. Pattern Recog. (2018)
85. Zhou, P., Xie, L., Ni, B., Tian, Q.: CIPS-3D: A 3d-aware generator of gans based on conditionally-independent pixel synthesis. arXiv preprint arXiv:2110.09788 (2021)
86. Zhu, J., Shen, Y., Zhao, D., Zhou, B.: In-domain GAN inversion for real image editing. In: Eur. Conf. Comput. Vis. (2020)
87. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: Eur. Conf. Comput. Vis. (2016)